AFRL-RI-RS-TR-2010-111

ADVANCED COMPUTING ARCHITECTURES FOR HIGH PERFORMANCE COMPUTING ENGINEERING INTEGRATION

Rome Research Corporation

May 2010

FINAL TECHNICAL REPORT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

STINFO COPY

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE

■ **AIR FORCE MATERIEL COMMAND** ■ **UNITED STATES AIR FORCE** ■ **ROME, NY 13441**

NOTICE AND SIGNATURE PAGE

AFRL-RI-RS-TR-2010-111 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/	/s/
DUANE A. GILMOUR	EDWARD J. JONES, Deputy Chief
Work Unit Manager	Advanced Computing Division
	Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
MAY 2010

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
February 2007 – February 2010

4. TITLE AND SUBTITLE
ADVANCED COMPUTING ARCHITECTURES FOR HIGH PERFORMANCE COMPUTING ENGINEERING INTEGRATION

5a. CONTRACT NUMBER
FA8750-07-C-0023

5b. GRANT NUMBER
N/A

5c. PROGRAM ELEMENT NUMBER
62702F

6. AUTHOR(S)
Edward Killian

5d. PROJECT NUMBER
558T

5e. TASK NUMBER
CL

5f. WORK UNIT NUMBER
US

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Rome Research Corporation
314 South Jay Street
Rome, NY 13440-5600

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/RITA
525 Brooks Road
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)
N/A

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TR-2010-111

12. DISTRIBUTION AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited. PA# 88ABW-2010-2726
Date Cleared: 19-May-2010

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Rome Research Corporation (RRC) performed system engineering, development and integration studies for the Air Force Research Laboratory Advanced Computing Division (AFRL/RIT) at the Rome Research Site (RRS) in the areas of Advanced Computing Architectures (ACA) and High Performance Computing (HPC) under contract number FA8750-07-C-0023. AFRL/RIT is responsible for researching and recommending advanced computing architectures in support of Command, Control, Communications, Computer, and Intelligence (C4I) and surveillance systems. The objective of the ACA for HPC effort was to apply state-of-the-art Information Technologies (IT) to C4I and surveillance systems.

15. SUBJECT TERMS
Advanced Computing Architectures, High Performance Computing

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
U	U	U	UU	29	Duane A. Gilmour
					19b. TELEPHONE NUMBER (*Include area code*) N/A

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

TABLE OF CONTENTS

LIST OF FIGURES

1 INTRODUCTION

Rome Research Corporation (RRC) performed system engineering, development and integration studies for the Air Force Research Laboratory Advanced Computing Division (AFRL/RIT) at the Rome Research Site (RRS) in the areas of Advanced Computing Architectures (ACA) and High Performance Computing (HPC) under contract number FA8750-07-C-0023. AFRL/RIT is responsible for researching and recommending advanced computing architectures in support of Command, Control, Communications, Computer, and Intelligence (C4I) and surveillance systems. The objective of the ACA for HPC effort was to apply state-of-the-art Information Technologies (IT) to C4I and surveillance systems.

1.1 Background

The mission of the AFRL ACA Core Technical Competency (CTC) is to explore and develop computer architectures with greater capacity, sophistication and assurance for addressing dynamic mission objectives under constraints imposed by C4, ISR, and strike systems. This includes new computational paradigms, which enable aerospace weapons systems to achieve global information dominance and aerospace superiority. The division advances the state-of-the-art in these sciences and technologies to produce capabilities that are not commercially available or mature enough for combat systems. Its further mission is to deliver new capabilities for air, space and cyberspace applications.

1.2 Scope

The scope of this effort included:

- Analyzing existing IT systems
- Recommending architecture enhancements and upgrades
- Applying updates
- Documenting configurations
- Testing architecture components

1.3 Document Purpose

This Final Technical Memorandum, Contract Data Requirements List (CDRL) item A007, provides a summary of the support and technical expertise provided by RRC in support of the AFRL program objectives.

1.4 Document Overview

This document has been organized into the following sections:

- ***Section 1*** captures the purpose and intent of this document and provides summary information relating to the effort.

- ***Section 2*** summarizes the project accomplishments during the duration of the project lifecycle.

- ***Section 3*** provides summary conclusions.

- ***Section 4*** presents recommendations for future Research and Development (R&D).

- ***Section 5*** provides a list of references.

- ***Appendix A*** presents a list of the acronyms appearing in this document.

- ***Appendix B*** is the Backup Procedures document delivered under the ACA For HPC effort.

- ***Appendix C*** contains scripts that were written to maintain and configure the HPC systems.

2 PROJECT ACCOMPLISHMENTS

2.1 Evaluation and Assessment Reporting

RRC staff members attended weekly Client Support Administrator (CSA) meetings to stay informed of system and network changes within RRS which could affect the R&D systems managed by the team. These discussions related to security settings, patches, and proposed changes for future implementation. Additional meetings specific to the Wireless Distributed Decision Architecture (WDDA) group were conducted. RRC was responsible for assessment of the infrastructure of WDDA systems and the operating system level software.

2.2 Investigation of State-of-the-Art Technologies

RRC staff members were responsible for the consistency of the day-to-day administrative and technical system performance, as well as resolution of problems reported by users. This applied to the R&D desktop systems as well as the HPC clusters.

2.3 Software and Hardware Component Selection

RRC staff members performed initial diagnostics and trouble-shooting on systems assigned to them by AFRL personnel. Additionally, RRC staff was responsible for formatting, partitioning, conducting backup procedures and restoring hard drives. The RRS Automated Data Processing Equipment (ADPE) custodian was notified of any hardware relocation. When requested, RRS ADPE and Tivoli Office personnel were assisted with computer hardware and software inventories.

2.4 Installation and Configuration

RRC staff was required to review daily system logs, comply with operating system and application patches or configurations, and be capable of installing and configuring all applicable client/server devices within their purview.

User's terminals, workstations and all appropriate Division and Subnet servers or resources were included as areas of responsibility. RRC staff members performed the installation of equipment, connection of peripherals, and the installation/deletion of user software, not including the network backbone infrastructure itself.

The Wireless Distributed Decision Architecture (WDDA) effort utilized Mobile Internet Devices (MID). These were preinstalled with MIDinux, a version of Linux customized for the MID. RRC installed and tested other versions of Linux to determine which version of Linux would run on the complete Casadable Collaborative Composite Processors (C3P) environment and on the MID. During this effort RRC customized the Linux installs on the MID to allow its special features to work. This included enabling a touch screen interface and getting the internal wireless interface to work in Ad-Hoc mode. Figure 1 depicts the checklist used for this installation task.

- [ ] Install Linux OS – careful selection of install packages
- [ ] Up date with current patches
- [ ] Install current banners in motd, issue, issue.net and make sure that motd doesn't get overwritten at boot
- [ ] Setup minimum password requirements and expiration
/etc/security/pam_pwcheck.conf
  - [ ] Minlen=8
  - [ ] Dcredit=-2
  - [ ] Ucredit =-2
  - [ ] Lcredit=-2
  - [ ] Ocredit=-2
  - [ ] Type=secure
- [ ] Check for world writable files – find / -perm -002 –type f –o type d -ls
- [ ] Modify /etc/inittab "ca:" entry to disable CTRL-ALT-DELETE
- [ ] Add sulogin to /etc/inittab "sp" entry to force root password on single user boot
- [ ] Configure logwatch
- [ ] Install Antivirus – Symantec preferred, ClamAV
Make sure AV update works
- [ ] Configure and schedule (cron) AV scans
- [ ] Crontab updates for AV scan, sendmail –q, etc.
- [ ] Configure email for sending mail
- [ ] Complete IA UNIX/Linux checklist
- [ ] Install and configure backup

Figure 1 – Example Installation Checklist

2.5 Back-up Management

RRC specified, installed and configured the backup system within the RIT division. This system was used to back up both the Office Automation (OA) network systems and the R&D network systems. Recent changes within the network infrastructure have forced this system to be moved to the R&D network and currently only backs up the R&D systems. RRC has specified both the hardware and software for a new backup system for the OA network.

The backup system's log files are checked daily to ensure backups are operating properly and to troubleshoot problems with the backups. Additional information regarding backup procedures is provided in Appendix B. Issues that have occurred with the current R&D backup system include tape library hardware problems and backup software updates/upgrades. The tape library hardware problem was resolved by installation of a new tape library. Also, the configuration of the backup system was set up to mimic the branch structure of the RIT division. When the division was reorganized, RRC reconfigured the backup system to match. There was a problem with backing up files that were open while the system was being backed up. RRC specified, installed, and configured software (Open File Manager) on each backup client to solve this problem. During 2009, RIT procured an Overland Storage Redundant Array of Independent Disks (RAID) system. RRC installed this onto the backup system and configured it as a virtual tape library for backup storage. This included installing and configuring a fiber channel interface both in the server and in the RAID system.

2.6 DAA Accreditation Process Compliance

RRC configured and modified user software configurations to comply with Designated Approving Authority (DAA) approved software configurations and performed basic configuration management functions.

A member of the RRC staff, in his capacity as Workgroup Manager (WM), was responsible for reporting security breaches, distributing security information, and assisting in the development and maintenance of the Systems Security Certification & Accreditation package used for network Certification and Accreditation (C&A). The WM obtained an implementation checklist from the National Cryptographic Command (NCC) and the Information Assurance (IA) Office before installing equipment and was responsible for assisting with installation, testing, and acceptance of the system according to the terms of the purchase contract and instructions. Additionally, RRC staff members assisted the NCC, the RRS Information System Security Officer (ISSO), and the Division Functional Systems Administrator (FSA) in implementing network security policies and procedures and complying with all tasks outlined in AFI 33-115v1, Chapter 4, Paragraph 4.8.

2.7 Operating Systems Research

Various Commercial-off-the-shelf (COTS), Government-off-the-shelf (GOTS), open source, and custom solutions were investigated. Research is designed to support command and control systems with shared computational capabilities as needed in ground combat situation where data and communications links may be marginal. In particular, the ACA HPC clusters were running the outdated SuSE 9.3 operating system. The requirement to be connected to a RRS network is that the operating system has to be supported for critical and security patches. This version of SuSE was not. A survey of acceptable operating systems was performed and Debian 5 Linux provided the technical solution.

Within the WDDA effort there was a need for the operating system as previously described. Before the decision was made to utilize the Debian 5 operating system, evaluations on several different versions of operating systems were performed to see which best fit the requirements. These operating systems included Fedora Versions 9 and 10, Mandriva, Ubuntu, and Debian Version 4 and 5.

The RIT division operated a 30-node cluster which was used to research parallel processing using the Message Passing Interface (MPI). The cluster was running an older version of the Red Hat operating system and needed to be updated. RRC installed the Fedora Core 5 operating system and Open Source Cluster Application Resource (OSCAR) Version 3.0.

2.8 WDDA

Within the WDDA effort, the operating system was required to be current, open source, main stream, and support the varied hardware that was being utilized. Again, Debian 5 Linux was used since it met all the criteria and it would be a common operating system with the ACA HPC clusters.

This project started as the Wireless Computational Network Architecture (WCNA) and was renamed within the last three months of the contract effort. Initially the project used a Linux distribution called Mandriva 2008.1. There were issues with this version of Linux concerning its support for the wireless networking the project is using. RRC tested different versions of Linux and selected Debian 5. RRC installed the operating system and configured the "head" node as a Dynamic Host Configuration Protocol (DHCP) and a Network File System (NFS) server. Because of special requirements of the WDDA project, RRC needed to modify the normal boot sequence. Special boot scripts were written and the DHCP server files were modified to supply special files to all the DHCP clients. Whichever node had a wireless network interface connected to it would be the Controlling Functional Block (CFB). If the system was a CFB, it would dynamically configure itself to be a DHCP server, NFS server, and controlling node via a script. On a system shutdown or reboot, the system would un-configure itself in case the wireless network was disconnected and connected to another node. Appendix C contains a sample of scripts developed to address this issue.

The WDDA enclave is primarily contained in the Naresky Lab Suite F8 in Building 3, but some work with the MIDs required expansion into various offices within Building 3. To increase data throughput, wireless network protocols were researched in the WDDA enclave. This enables distributed decision making to support command and control. WDDA is also used to research and develop systems that demonstrate the inherent power available in small form factor compute devices consisting of multiple processors working cooperatively and/or independently.

The WDDA standalone network enclave consists of several elements. These include MID devices, C3P systems, a regular desktop system, and relay nodes. The MID devices communicate wirelessly to each other as well as to the C3P systems. The desktop system was used for a development platform as well as a software repository. Relay nodes not only relay traffic but also provide computational functionality. The C3P are picoITX-based module computational clusters which communicate internally via wired network and externally via wireless. A Global Positioning System (GPS) was incorporated for location information as well as time and date information. Research is oriented towards command and control systems with shared computational capabilities as needed in ground combat situation where data and communications links may be marginal. Figure 2 provides a functional block diagram of the C3P.

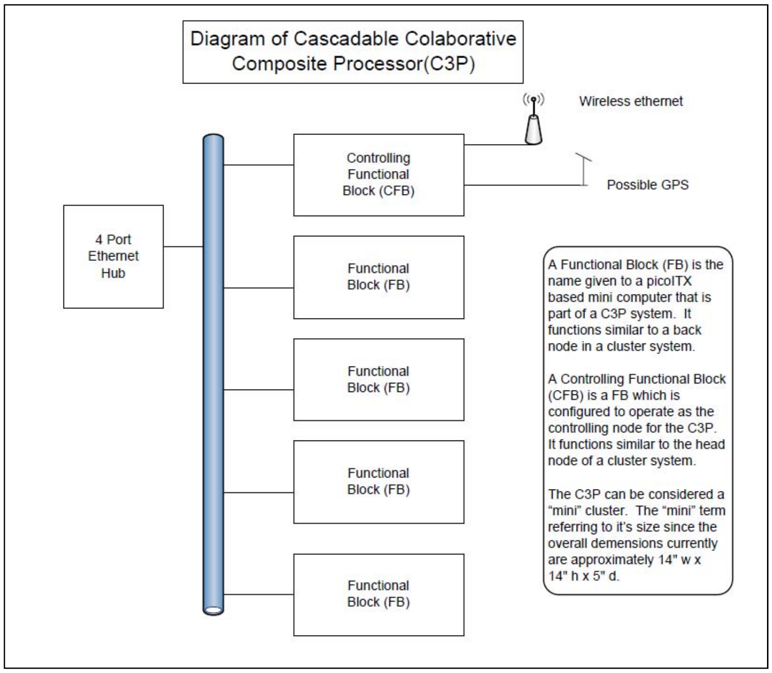

Figure 2 – Functional Block Diagram of the C3P

2.8.1 Testbed

RRC ran several tests to evaluate several configurations, including the following:

- One C3P communicating only wirelessly.
- Two C3Ps communicating completely wirelessly.
- One C3P communicating via hardwire and two C3Ps communicating internally using hardwire and externally using wireless.

These tests showed that the wireless network communication was the limiting factor. The configuration of two C3Ps connected together via hardwire was also tested in a cascaded form.

2.9 HPC Clusters

During the effort, RRC personnel supported the stand up and use of 3 HPC clusters. The clusters had various configurations of Central Processing Units (CPUs), number of nodes, and networking. They were used in support of ACA research projects as well as to evaluate cluster configurations for use in high performance computing.

The Latte cluster was a 30 node cluster running the Fedora Core 5 Linux operating system. The Latte cluster was also running the OSCAR 3.0 open source cluster application resource software package. The cluster had one head node with 30 compute nodes. The head node had two connected network ports, one connected to the RRS network and the other connected to a high speed network switch for intercluster communication. Latte utilized an NFS server that all the compute nodes used, to mount home directories which made compute data available. The cluster was best used with the install of MPI software but could be used as a "cluster of nodes" instead of an integrated cluster. Figure 3 provides a picture of the Latte head node.

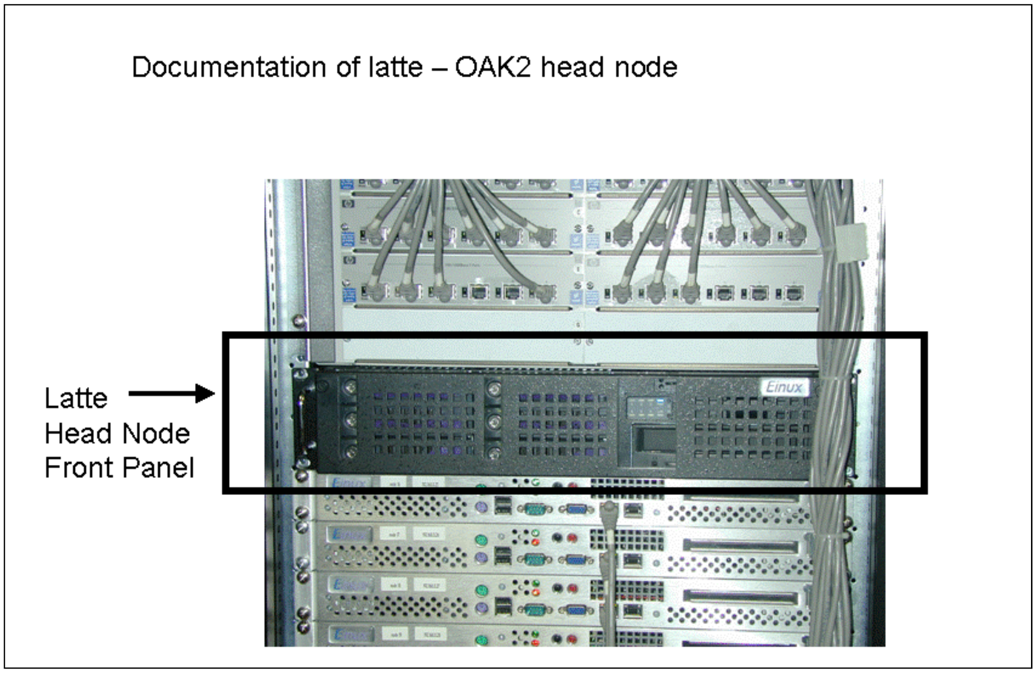

Figure 3 – Latte Head Node

The Latte cluster was the oldest cluster that RIT owned and had an older, slower CPU in the compute nodes. It was decommissioned and the hardware turned in to the ADPE custodians in December 2008.

AFRL/RIT currently has two other active clusters which have both been recently updated to Debian 5.0 and are running the standard MPICH Debian package. These are compute clusters but also may be used as a "cluster of nodes". One cluster has 16 compute nodes and the other has 26 nodes. Each node of the 16 node cluster has two dual core AMD Opteron processors for a total of 64 cores. Each node of the 22 node cluster also contains two dual core AMD Opteron processors for a total of 88 cores. The 16 node cluster's CPUs run at 2.4 GHz while the 22 node cluster runs at 2.2 GHz.

3 CONCLUSIONS

This Final Technical Memorandum was prepared by recapping various technical discussions, electronic mails, monthly status reports, contractor presentations, and other documentation maintained during the course of this effort. The objective of this effort was to design, develop, integrate, test, and evaluate architectures for High Performance Computing applications. As technology progresses, testbeds such as these will provide important feedback and input to advanced computing architectures.

4 RECOMMENDATIONS

Recent changes within the network infrastructure have forced the backup system to be moved to the R&D network and currently only backs up the R&D systems. RRC specified both the hardware and software for a new backup system for the OA network. Installing and configuring this system will be necessary after this contract has been completed.

It is recommended that the boot process of the nodes within the WDDA environment be documented, specifically the files that were modified.

It is further recommended that the 22 node cluster and the 26 node cluster be documented, including the operating system and the hardware used in the clusters.

5 REFERENCES

1. Air Force Instructions 33-115 Volume 1, *Network Operations* Chapter 4, Paragraph 4.8.
2. DoD 8570.01, *Information Assurance Training, Certification, and Workforce Management*

APPENDIX A ACRONYM LIST

Acronym	Definition
ACA	Advanced Computing Architectures
ADPE	Automated Data Processing Equipment
AFRL	Air Force Research Laboratory
AMD	Advanced Micro Devices
C&A	Certification and Accreditation
C3P	Cascadable Collaborative Composite Processors
C4I	Command, Control, Computer, and Intelligence
CDRL	Contract Data Requirements List
CFB	Controlling Functional Block
COTS	Commercial-off-the-Shelf
CPU	Central Processing Unit
CSA	Client Support Administrator
CTC	Core Technical Competency
DAA	Designated Approving Authority
DHCP	Dynamic Host Configuration Protocol
GOTS	Government-off-the-Shelf
GPS	Global Positioning System
HPC	High Performance Computing
IA	Information Assurance
ISSO	Information System Security Officer
IT	Information Technologies
MID	Mobile Internet Devices
MPI	Message Passing Interface
NCC	National Cryptographic Command
NFS	Network File System
OA	Office Automation
OSCAR	Open Source Cluster Application Resource
R&D	Research and Development
RAID	Redundant Array of Independent Disks
RRC	Rome Research Corporation
RRS	Rome Research Site
SSAA	Systems Security Certification & Accreditation
WCNA	Wireless Computational Network Architecture
WDDA	Wireless Distributed Decision Architecture
WM	Workgroup Manager

APPENDIX B ACA FOR HPC BACKUP PROCEDURES

B.1 Overview

This document describes the ACA for HPC backup procedures. Included are backup server hardware configuration, backup server software configuration, and client configuration. Also included are daily, monthly and yearly tasks which need to accomplished for proper backup completion.

B.2 Backup Server Hardware Configuration

B.2.1 Server

The backup server is a Sun Enterprise V240.

B.2.2 Tape Library

The backup tape drive is a Quantum Superloader LTO tape drive with one internal LTO tape drive and two 8 slot magazines.

B.2.3 Virtual Tape Library

The virtual tape library is an Overland Storage REO 4500 configured as an autochanger and one tape drive. The tape drive emulates an Ultrium LTO tape drive. The library is configured with 30 slots. Each slot contains a virtual tape which has a capacity of 300 GB.

B.3 Backup Server Software Configuration

B.3.1 Server Operating System

The backup server is running Solaris 10.

B.3.2 Server Backup Software

- The backup software is Sun StorageTek Enterprise Backup Software (EBS)
- Client licenses
- Modules

B.4 Client Configuration

B.4.1 Client Software

The backup clients are running Sun StorageTek client software.

B.5 Daily Tasks

B.5.1 Group Backup Email Checks.

As each backup group finishes its daily run, an email which contains the results of each client within the group is generated and sent to the configured email address. These emails can be broken down into two types. The first type is when all clients within the group finish successfully and the second type is when at least one client fails. If a backup group has not finished for some reason, an email will not be sent until the backup has finished for that group. This group should be checked to see if there is a reason for the non-completion.

B.5.2 Successful

Successful group emails are when all clients within the group complete their backup successfully. When reviewing this type of email, no other action is needed. An example email is shown below.

```
From:        Super-User
Sent:        Sunday, February 08, 2009 9:08 PM
To:
Subject:     back33's savegroup completion

Sun StorageTek(TM) Enterprise Backup savegroup: (notice) RIT RD  NG completed, Total 2
client(s), 2 Succeeded. Please see group completion details for more information.

Succeeded: killian2, spetka2

Start time:   Sun Feb  8 19:33:00 2009
End time:     Sun Feb  8 21:08:25 2009


--- Successful Save Sets ---

* killian2:All savefs killian2: succeeded.
  killian2: /                   level=incr,     24 MB 00:00:07    146 files
  back33: index:killian2        level=9,        64 KB 00:00:02      6 files

* spetka2:All savefs spetka2: succeeded.
  spetka2: /boot                level=incr,      0 KB 00:00:03      0 files
  spetka2: /                    level=incr,    181 MB 00:32:52    345 files
  back33: index:spetka2         level=9,       122 KB 00:00:02      6 files
```
e

Figure 4 – Successful group email

B.5.3 Failed

A failed group email is one in which at least one of the clients in the group has failed to complete a successful backup. When reviewing this type of email, the reason for the failure must be determined and if possible remediated. An example email is shown below.

```
From:          Super-User
Sent:          Sunday, February 08, 2009 9:40 PM
To:
Subject:       back33's savegroup completion


Sun StorageTek(TM) Enterprise Backup savegroup: (alert) RITM completed, Total 6 client(s), 1
Failed, 5 Succeeded. Please see group completion details for more information.

Failed: sd35106367
Succeeded: sda7061bxd, sda7430k84, sda81402rb, sda8310987, sda8360w24


saveset sda8310987:C:\Documents and Settings: percentage of inactive files by count: 1.29, by
space: 2.21


saveset sda81402rb:C:\Documents and Settings: percentage of inactive files by count: 53.17,
by space: 20.72


saveset sda8360w24:C:\Documents and Settings: percentage of inactive files by count: 9.36, by
space: 18.96

Start time:   Sun Feb  8 19:33:00 2009
End time:     Sun Feb  8 21:39:58 2009


--- Never Started Save Sets ---

savegrp: sd35106367:C:\Documents and Settings save was never started
* Probe job had unrecoverable failure(s), this job is being abandoned.


--- Unsuccessful Save Sets ---

* sd35106367:Probe logfile has no output
    * <SEVERE> :  Connection timed out


--- Successful Save Sets ---

  back33: index:sd35106367          level=9,         2 KB 00:00:02      2 files

* sda7061bxd:Probe savefs sda7061bxd: succeeded.
  sda7061bxd: C:\Documents and Settings level=incr, 82 MB 00:01:03     84 files
```

Figure 5 – Failed group email

There are many possible causes of a failed backup which include, but are not limited to, network connection problems, laptops not on the network, DNS failures or errors, and configuration errors. Each client that fails needs to be diagnosed and either remediated or, in certain cases, ignored. The only case that can be ignored currently is the case where a client is a laptop and the user is TDY with the laptop.

B.5.4 Daily Printout

After the backup of the backup server has completed the backup software will generate and print a report of the backup server's bootstrap information. The information contained in the report is important since it is needed to restore the backup server if it has a hard disk failure. This report is collected from the printer and then stored in the appropriate area for reference if needed. A portion of a bootstrap report is shown below.

```
February 11 18:37 2009  back33's bootstrap information Page 1

      date      time       level ssid          file  record   volume
       01/07/09 22:45:30    full 3345315546     28        0   000048L3
       01/08/09 22:37:19    full 644269680      94        0   000048L3
       01/12/09 14:43:56    full 2171313922     59        3   000059L3
       01/12/09 22:36:15    full 3513519151    112        0   000059L3
       01/13/09 22:42:09    full 728588049      50        0   000052L3
       01/14/09 22:40:25    full 1500426281      5        0   000053L3
       01/15/09 22:37:36    full 3044016384     72        0   000053L3
       01/20/09 22:50:00    full 460757864      32        0   000054L3
       01/21/09 22:47:38    full 1618472026    105        0   000054L3
       01/22/09 22:50:17    full 3094953593     51        0   000055L3
       01/23/09 22:56:31    full 343576943       7        0   000056L3
```

Figure 6 – Bootstrap printout

The procedure to restore the backup server if there is a hard disk failure can be found in the appropriate EBS manual. The basic sequence would be to replace the hard drive, reinstall the system OS, reinstall the backup software so that it can read the backup tapes, and then use the bootstrap information to restore the backup saveset.

B.6 Monthly Tasks

B.6.1 Server Full Backup to Tape

Even though the backup system can be restored from the backup tapes exclusively, the backup server should be backed up using the standard UNIX dump command monthly. This will not only save time to restore the backup server but will also make it easier. When there is sufficient time to complete the task the server should be taken down to single user mode. Then, using the dump command, the system should be backed up to a separate tape that is designated for this purpose only. In this way, instead of reinstalling the OS, reconfiguring the system and reinstalling the backup software, the system can be restored to the state it was in at the time of this backup using only one tape and standard UNIX commands.

B.6.2 Full backup Saveset Clones.

With the recent addition of the virtual tape library (VTL), there is a need monthly to copy, or clone, the full backup save sets to physical tape for archival purposes. This procedure is still being developed.

APPENDIX C CUSTOM SCRIPTS

Contained in this appendix are sampling of the scripts developed to maintain and configure the HPC systems.

C.1 pico-init

```
#!/bin/sh
#
#       Author: Ed Killian
#

### BEGIN INIT INFO
# Provides:          pico-init
# Required-Start:
# Required-Stop:
# Default-Start:     S
# Default-Stop:      0 6
# Short-Description: Configures pico system (WCNA)
### END INIT INFO


PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

DESC="Configures system as a CFB (WCNA)"
CFBCONF=/etc/cfb/cfb.conf
SELF=$(cd $(dirname $0); pwd -P)/$(basename $0)
INTERFACESWIRELESS=/etc/cfb/interfaces-wireless
INTERFACES=/etc/network/interfaces
DHCPCONF=/etc/dhcp3/dhcpd.conf
DHCPCONFSAMPLE=/etc/cfb/dhcpd.conf
DHCPSERVERFILE=/etc/default/dhcp3-server
DHCPSERVERSAMPLE=/etc/cfb/dhcp3-server
NTPCONFSERVER=/etc/cfb/ntp.conf-server
NTPCONFCLIENT=/etc/cfb/ntp.conf-client
NTPCONF=/etc/ntp.conf
HOSTNAME_FILE=/etc/hostname
MAILNAME_FILE=/etc/mailname
OLSRDCONF=/etc/olsrd/olsrd.conf
OLSRDCONFSAMPLE=/etc/cfb/olsrd.conf
SSH_KNOWN_HOSTS=/etc/ssh/ssh_known_hosts
GMONDCONF=/etc/ganglia/gmond.conf
GMONDCONFSAMPLE=/etc/cfb/gmond.conf
GMETADCONF=/etc/ganglia/gmetad.conf
GMETADCONFSAMPLE=/etc/cfb/gmetad.conf

BASEDIR=/

. /lib/lsb/init-functions

#
# Make sure required files are there
```

```
#
#if [ ! -x $REQUIRED ]; then
#       log_failure_msg "gpsd: error: Cannot find $REQUIRED."
#       exit 1
#fi

dummy=`udevinfo -a -p /sys/class/net/eth1 | grep Wireless > /dev/null`
WIRELESS_PRESENT=$?

case $1 in
  start)
        # Check if $WIRELESS present is 0, then WIRELESS is present and it's
CFB
        # if $WIRELESS is NOT 0, then WIRELESS is not present and it's NOT
CFB.
        if [ $WIRELESS_PRESENT -eq 0 ] ; then
           if [ ! -e $CFBCONF ]; then
                echo "WIRELESS=XXX.XXX" > $CFBCONF
           fi

           . $CFBCONF

           if [ $WIRELESS == "XXX.XXX" ]; then
              log_daemon_msg "CFB unconfigured - running configure."
              log_end_msg 0

              WIRELESS_IP=0
              GOOD_IP=0

              while [ $GOOD_IP -eq 0 ]; do

                whiptail --inputbox IP --clear --title "Enter last two octets
of wireless IP" 5 45 "3." 2> /tmp/whiptail.$$
                TMP=`cat /tmp/whiptail.$$`
                rm -f /tmp/whiptail.$$
                TMP=${TMP:-00}

                echo $TMP | grep "\." > /dev/null 2>&1
                RES=$?

                if [ $RES -eq 1 ]; then
                  TMP="3."$TMP
                fi

                THIRD_OCTET=`echo $TMP | awk -F. '{print $1}'`
                FOURTH_OCTET=`echo $TMP | awk -F. '{print $2}'`

                FOURTH_OCTET=${FOURTH_OCTET:-0}

                if [ $THIRD_OCTET -gt 0 -a $THIRD_OCTET -lt 255 ]; then
                  TH_GOOD=1
                else
                  TH_GOOD=0
                fi
```

```
    if [ $FOURTH_OCTET -gt 0 -a $FOURTH_OCTET -lt 255 ]; then
      FO_GOOD=1
    else
      FO_GOOD=0
    fi

    if [ $TH_GOOD -eq 1 -a $FO_GOOD -eq 1 ]; then
      GOOD_IP=1
    fi

  done

  WIRELESS_IP=$THIRD_OCTET"."$FOURTH_OCTET

  echo "WIRELESS="$WIRELESS_IP > $CFBCONF

  . $CFBCONF

fi

log_daemon_msg "Starting FB configuration" && log_end_msg 0

#
# If cfb, copy ntp.conf for server
#
sed -e 's/XXX.XXX/'$WIRELESS'/g' $NTPCONFSERVER > $NTPCONF

log_daemon_msg "NTP configured" && log_end_msg 0
#
# Configure wireless network and wired
#
sed -e 's/XXX.XXX/'$WIRELESS'/g' $INTERFACESWIRELESS > $INTERFACES
log_daemon_msg "Networks configured" && log_end_msg 0

# Now we need to set the DHCP file /etc/dhcp3/dhcpd.conf for IP
# Take default file and edit
sed -e 's/XXX.XXX/'$WIRELESS'/g' $DHCPCONFSAMPLE > $DHCPCONF
cp $DHCPSERVERSAMPLE $DHCPSERVERFILE
log_daemon_msg "DHCP configured" && log_end_msg 0

# Now we need to set the olsrd file /etc/olsrd/olsrd.conf for IP
# Take default file and edit
#sed -e 's/XXX.XXX/'$WIRELESS'/g' $OLSRDCONFSAMPLE > $OLSRDCONF
log_daemon_msg "OLSRD configured" && log_end_msg 0

# CFB servers NFS directory
sed -e 's/XXX.XXX/'$WIRELESS'/g' /etc/cfb/exports > /etc/exports
log_daemon_msg "/etc/exports configured" && log_end_msg 0


# If cfb, set hostname - non CFB uses dhcp to set hostname
#
```

```
          # OK, start the magic here.  Basically we want to set the hostname
to be
          # picoAAABBBCCC with the AAA being a zero padded entry of the
second octet
          # and BBB being a zero padded entry of the third octet and CCC
being a zero
          # padded entry of the last octet of the IP address passed by DHCP.
          # (i.e. pico003022006 for IP address 10.3.22.6 or
          # pico003022012 for IP 10.3.22.12 or pico003022145 for IP address
10.3.22.145)

          zeropadd3=`echo $WIRELESS | awk -F. '{printf("%.3d", $1)}'`
          zeropadd4=`echo $WIRELESS | awk -F. '{printf("%.3d", $2)}'`
          last_octet=`echo 5`
          zeropadded=`echo $last_octet | awk '{printf("%.3d", $0)}'`
          hostname=pico$zeropadd3$zeropadd4$zeropadded

          echo $hostname > $HOSTNAME_FILE
          echo $hostname > $MAILNAME_FILE

          /etc/init.d/hostname.sh start


          # Build hosts file
          lower_end=`grep -v "^#" /etc/dhcp3/dhcpd.conf | grep range | awk
'{print $2}' | awk -F. '{print $4}'`
          upper_end=`grep -v "^#" /etc/dhcp3/dhcpd.conf | grep range | awk
'{print $3}' | awk -F. '{print $4}'| tr -d ';'`

          echo "127.0.0.1     localhost" > /etc/hosts
          echo "10."$WIRELESS".5      "$hostname" picoCFB" >> /etc/hosts
          cnt=$lower_end
          while [ $cnt -le $upper_end ]; do

               zeropadded=`echo $cnt | awk '{printf("%.3d", $0)}'`
               echo "10."$WIRELESS"."$cnt"
       pico"$zeropadd3$zeropadd4$zeropadded >> /etc/hosts
               cnt=`expr $cnt + 1 `
          done
          log_daemon_msg "/etc/hosts configured" && log_end_msg 0

          # Build ssh_known_hosts file
          echo $hostname",picoCFB,10."$WIRELESS".5 ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAzNUmYkjM+wlH70KEXXPv5OcaNhPIyc8p07h5WxrlDx3XaB59d
ldeGRGkBcfrlYJUb+dPZaJRyfvnSCyehZ1xt+7mMSpFnIgF8aYMnhLFbANpqV2+AC0k1ef2uaYiQI
lOap+yznb5DXQAAP3H4pOHQCRkLWR/g2PL3qMkqSyK0NE3AZE1HmmkCOBsBWEYw7M7ZIQZIkYzjBn
ul4Bepa93F+7TmYoJ/Mj6SYfBxllAoBAob0Fg3rVeg3U/QZPbvO+2aIyGsB6odAv9ELOR2bBezVy6
RM6uj+I1KsiQW1Na76kVE+3gr2ZPQ4qVIeGSgrvH5beSecRHmbP9lhsElDLMow==" >
$SSH_KNOWN_HOSTS
          cnt=$lower_end
          while [ $cnt -le $upper_end ]; do

               zeropadded=`echo $cnt | awk '{printf("%.3d", $0)}'`
```

```
                echo
"pico"$zeropadd3$zeropadd4$zeropadded",10."$WIRELESS"."$cnt" ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAzNUmYkjM+wlH70KEXXPv5OcaNhPIyc8p07h5WxrlDx3XaB59d
ldeGRGkBcfrlYJUb+dPZaJRyfvnSCyehZ1xt+7mMSpFnIgF8aYMnhLFbANpqV2+AC0k1ef2uaYiQI
lOap+yznb5DXQAAP3H4pOHQCRkLWR/g2PL3qMkqSyK0NE3AZE1HmmkCOBsBWEYw7M7ZIQZIkYzjBn
ul4Bepa93F+7TmYoJ/Mj6SYfBxllAoBAob0Fg3rVeg3U/QZPbvO+2aIyGsB6odAv9ELOR2bBezVy6
RM6uj+I1KsiQW1Na76kVE+3gr2ZPQ4qVIeGSgrvH5beSecRHmbP9lhsElDLMow==" >>
$SSH_KNOWN_HOSTS
                cnt=`expr $cnt + 1 `
           done
           log_daemon_msg "/etc/ssh/ssh_known_hosts configured" && log_end_msg
0

           # Set the CLUSTERNAME in the GMETADCONF file
           #
           CLUSTERNAME="C3P"$zeropadd3$zeropadd4
           sed -e 's/data_source "my cluster"/data_source "'$CLUSTERNAME'"/'
$GMETADCONFSAMPLE > $GMETADCONF
           log_daemon_msg $GMETADCONF "configured" && log_end_msg 0

           # Set the CLUSTERNAME in the GMONDCONF file on CFB
           #
           sed -e 's/name = "unspecified"/name = "'$CLUSTERNAME'"/'
$GMONDCONFSAMPLE > $GMONDCONF
           log_daemon_msg $GMONDCONF "configured" && log_end_msg 0

        else
           log_failure_msg "Cannot find Wireless - system will not be
configured as CFB" && log_end_msg 0

           # Not a CFB so do not configure NFS, DHCP, or wireless.  Set for
dhcp client.
           if [ -e /etc/exports ]; then
                rm -f /etc/exports
           fi
           if [ -e /etc/default/dhcp3-server ]; then
                rm -f /etc/default/dhcp3-server
           fi

           cp /etc/cfb/interfaces-dhcp $INTERFACES

           #
           # If not cfb, copy ntp.conf for client
           #
           cp $NTPCONFCLIENT $NTPCONF

           #
           # If not cfb, do not start gmetad
           #
           if [ -e /etc/ganglia/gmetd.conf ]; then
                rm -f /etc/ganglia/gmetd.conf
           fi

        fi
```

```
        ;;
  stop)
        # On shutdown, clean up system pico configuration
        # clean up everything, CFB or not

        log_daemon_msg "Cleaning up FB configuration" && log_end_msg 0

        cp /etc/cfb/70-persistent-net.rules /etc/udev/rules.d/70-persistent-
net.rules
        cp /dev/null $NTPCONF
        if [ -e /etc/ntp.conf.dhcp ]; then
           rm -f /etc/ntp.conf.dhcp
        fi
        cp /dev/null $DHCPCONF
        if [ -e /etc/default/dhcp3-server ]; then
           rm -f /etc/default/dhcp3-server
        fi
        cp /dev/null $OLSRDCONF
        cp /dev/null $SSH_KNOWN_HOSTS
        cp /dev/null $INTERFACES
        if [ -e /etc/exports ]; then
                rm -f /etc/exports
        fi
        echo "unconfigured" > $HOSTNAME_FILE
        echo "unconfigured" > $MAILNAME_FILE
        echo "127.0.0.1 localhost" > /etc/hosts

        if [ -d /var/lib/dhcp3 ]; then
           rm -f /var/lib/dhcp3/*
           cp /dev/null /var/lib/dhcp3/dhcpd.leases
        fi
        if [ -e /etc/ganglia/gmetad.conf ]; then
                rm -f /etc/ganglia/gmetad.conf
        fi
        if [ -e /etc/ganglia/gmond.conf ]; then
                rm -f /etc/ganglia/gmond.conf
        fi

        ;;
  restart)
        $SELF stop
        $SELF start
        ;;
  *)
        echo "Usage: $0 {start|stop|restart}" >&2
        exit 1
        ;;
esac

exit 0
```

C.2 sethost_name

```
HOSTNAME_FILE=/etc/hostname
MAILNAME_FILE=/etc/mailname
HOSTS_FILE=/tmp/hosts

#echo $reason >> /tmp/debugger

set_hostname_setup_remove() {
        if [ -e $HOSTNAME_FILE ]; then
                cp /dev/null $HOSTNAME_FILE > /dev/null 2&>1
        elif [ -e $MAILNAME_FILE ]; then
                cp /dev/null $MAILNAME_FILE > /dev/null 2&>1
        fi

        return

}


set_hostname_setup_add() {
        if [ -e $HOSTNAME_FILE ] && [ "$new_ip_address" = "$old_ip_adress" ];
then
                return
        fi

        #echo $new_ip_address >> /tmp/debugger

        if [ -z "$new_ip_address" ]; then
                set_hostname_setup_remove
                return
        fi

        # OK, start the magic here.  Basically we want to set the hostname to
be
        # picoAAABBBCCC with the AAA being a zero padded entry of the second
octet
        # and BBB being a zero padded entry of the third octet and CCC being
a zero
        # padded entry of the last octet of the IP address passed by DHCP.
        # (i.e. pico003022006 for IP address 10.3.22.6 or
        # pico003022012 for IP 10.3.22.12 or pico003022145 for IP address
10.3.22.145)

        zeropadd2=`echo $new_ip_address | awk -F. '{printf("%.3d", $2)}'`
        zeropadd3=`echo $new_ip_address | awk -F. '{printf("%.3d", $3)}'`
        zeropadd4=`echo $new_ip_address | awk -F. '{printf("%.3d", $4)}'`
        hostname=pico$zeropadd2$zeropadd3$zeropadd4

        echo $hostname > $HOSTNAME_FILE
        echo $hostname > $MAILNAME_FILE

        /etc/init.d/hostname.sh start
```

```
        # And while we have the information, go ahead and set CLUSTERNAME in
gmond.conf
        GMONDCONFSAMPLE=/etc/cfb/gmond.conf
        GMONDCONF=/etc/ganglia/gmond.conf
        CLUSTERNAME="C3P"$zeropadd2$zeropadd3
        sed -e 's/name = "unspecified"/name = "'$CLUSTERNAME'"/'
$GMONDCONFSAMPLE > $GMONDCONF

        /etc/init.d/ganglia-monitor restart

}


set_hostname_setup() {
        case $reason in
                BOUND|RENEW|REBIND|REBOOT)
                        set_hostname_setup_add
                        ;;
                EXPIRE|FAIL|RELEASE|STOP)
                        set_hostname_setup_remove
                        ;;
        esac
}


set_hostname_setup
```

C.3 zzget_known_hosts

```
KNOWN_HOSTS_FILE=/etc/ssh/ssh_known_hosts
DHCPSERVER=`grep dhcp-server-identifier /var/lib/dhcp3/dhclient.eth0.leases |
sed '$!d' | awk '{print $3}' | tr -d ';'`

#echo $reason >> /tmp/debugger

get_known_hosts_remove() {

        return

}


get_known_hosts_add() {

        # First build simple known_hosts file for DHCP server
        echo $DHCPSERVER" ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAzNUmYkjM+wlH70KEXXPv5OcaNhPIyc8p07h5WxrlDx3XaB59d
ldeGRGkBcfrlYJUb+dPZaJRyfvnSCyehZ1xt+7mMSpFnIgF8aYMnhLFbANpqV2+AC0k1ef2uaYiQI
lOap+yznb5DXQAAP3H4pOHQCRkLWR/g2PL3qMkqSyK0NE3AZE1HmmkCOBsBWEYw7M7ZIQZIkYzjBn
ul4Bepa93F+7TmYoJ/Mj6SYfBxllAoBAob0Fg3rVeg3U/QZPbvO+2aIyGsB6odAv9ELOR2bBezVy6
RM6uj+I1KsiQW1Na76kVE+3gr2ZPQ4qVIeGSgrvH5beSecRHmbP9lhsElDLMow==" >
$KNOWN_HOSTS_FILE
```

```
        scp $DHCPSERVER:$KNOWN_HOSTS_FILE $KNOWN_HOSTS_FILE

}


get_known_hosts() {
        case $reason in
                BOUND|RENEW|REBIND|REBOOT)
                        get_known_hosts_add
                        ;;
                EXPIRE|FAIL|RELEASE|STOP)
                        get_known_hosts_remove
                        ;;
        esac
}

get_known_hosts
```